

June 10, 2015

I. Strings

1. 'a'
2. '0'
3. '3'
4. IndexError: string index out of range
5. '...'
6. '2:9'
7. '0:43'
8. 'a0:12:90:00:8'
9. 'a19084'
10. my_string[6:11]
11. my_string[-11:-6]
12. my_string.count(':')
13. my_string.replace(':', '') **Remember:** You are actually replacing the colon with an empty str!
14. my_string.split(':')
15. ''.join(my_string.split(':'))

II. Numbers

1. Either 5 ** 3 or pow(5, 3).
2. 6
3. True
4. 3 (**Remember:** Python 2.x rounds down to the nearest integer when the result is a float!)
5. 3.3333333333333335
6. 11 (**Remember:** this the distance between the number we give as argument and 0!)
7. 'y' (**Remember:** Strings are considered to be larger than any numerical value!)

June 10, 2015

III. Booleans

1. False
2. False
3. True
4. False
5. True
6. True

IV. Lists

1. False
2. 10
3. [10, 'x', 20.02, 'y', 30j, 'z', 10L, False] (**Remember:** [:] returns the entire sequence!)
4. 'z'
5. ['y', 30j, 'z']
6. [30j, 'z']
7. [10, 'y', 10L]
8. [10, 'x', 20.02, 10, 'x', 20.02, 10, 'x', 20.02, 10, 'x', 20.02, 10, 'x', 20.02]
9. long
10. bool
11. my_list.append('new element')
12. a) del my_list[1] b) my_list.pop(1) c) my_list.remove('x')
13. my_list.index(20.02)
14. a) my_list.sort() b) sorted(my_list)
15. a) my_list.reverse() b) sorted(my_list, reverse = True)

June 10, 2015

V. Sets

1. `set1.add(500)`
2. `set1.remove(7)`
3. `set1.pop()`
4. `set1.intersection(set2)`
5. `set1.difference(set2)`
6. `set1.union(set2)`
7. `set2.clear()`

VI. Tuples

1. `'b'`
2. `1`
3. `(1, 2)`
4. `'a'`
5. `('a', 'b')`
6. `(3, 'a')`
7. `(1, 3, 'b', [4, 5, 6])`
8. `[4, 5, 6, 4, 5, 6, 4, 5, 6, 4, 5, 6, 4, 5, 6]`
9. `2`
10. `10`

VII. Dictionaries

1. `my_dict[6] = 'Nortel'`
2. `del my_dict['Nortel']`
3. `4 in my_dict`
4. `False`

June 10, 2015

5. 5

6. 'Cisco'

7. 'Avaya'

VIII. If/For/While/Nesting

1. 2 4 6 8

2. 11 22 33 44

3. 10000000000

IX. Regular Expressions

1. <type 'NoneType'>

2. '200.10.2.0 255.255.255.0 200.20.5.2 1 205 T#1 S IB 5'

3. '200.10.2.0'

4. '5'

5. '255'

6. '255'

7. '0'

8. '200.20.5.2 ' (the IP address followed by 6 spaces)

9. '205'

10. 'T'

11. ' '

12. 'I'

13. '5'

14. '200.10.2.0 255.255.255.0 200.20.5.2 1 205 T#1 S IB 5'

15. IndexError: no such group

16. 'Ether'

June 10, 2015

17. 'Switch'
18. '3549'
19. 'PWR+'
20. 'Ethernet Routing Switch 5xy5xy5xy5xyGTS-PWR+'

X. Advanced Python Tools

1. `[x * 10 for x in range(1, 11)]`
2. `[x * 10 for x in range(1, 11) if x <= 5]`
3. `[x * y for x in range(1, 6) for y in range(10, 13)]`
4. `[x * y for x in range(1, 6) for y in range(10, 13) if y <= 11]`
5. `my_lam = lambda x, y: x * y`
6. `my_lam = lambda x, y, z: x + " " + y + " " + z`
7. `my_lam = lambda dict: sorted(dict.keys(), reverse = True)`
8. `map(lambda x: x / 100, range(0, 1000, 100))`
9. `filter(lambda x: x % 100 == 1, range(0, 1000, 100))`
10. `reduce(lambda x, y: x * y, range(1, 11))`